Event-Driven Communication
User Guide
IGSS Version 8.0

Table of Contents

Chapter 1: Introducing Event-Driven Communication

1.1 Introduction

The purpose of this document is to describe the event-oriented communication procol used by the IGSS system and especially to document the layout/format of telegrams used by event-oriented communication.

The basic idea behind event-oriented communication is to let the external equipment (normally PLCs) send unsolicited data to the IGSS system. This means that the IGSS system is constantly listening for any data being send from external equipment thus reducing the communication overhead associated with traditional data polling. Event-oriented communication can be combined with traditional data polling in order to provide maximum flexibility by use of the event-oriented protocols and at the same time "keeping it simple" with traditional cyclic polling.

Notes

- Throughout this document, the layout of bytes is shown as it is represented in the PLCs, i.e. with the least significant byte on the left. This is the opposite of the PC convention.
- Data sent from the PLC to IGSS should be sent in chronologically correct sequences, i.e. an object should never receive data older than the current object value.

Chapter 2: Topology and Addressing

2.1 Addressing

The event-oriented protocol is originally designed for use with Simatic PLC equipment. This means that the addressing method used is designed to fit to this type of PLCs – but other PLC types and protocols might also be used. Please look into the documentation for the individual driver to get detailed information on how to use event-oriented communication.

The addressing of data values are as follows (note that an integral number of 16-bit words must always be transferred):

A number of data words in any ordinary data block in a direct node is addressed via the following components:

- Node number 0..255
    - Data Block 0..255
    - Word offset 0..65535
    - Word length 0..255

Chapter 3: Message Protocols and Format

3.1 Message Protocols and Format

This section describes the standard data transfer protocols used, and specifies the message format supported.

Data objects that are configured for direct transfer have their values transferred in the backward compatible mode, i.e. the node number is implied by the connection, and the data block, word offset, and word length are part of the standard protocol message header. The message body just contains the object data values corresponding to the addressed DB area.

Data objects configured for indirect event oriented transfer use the standard protocols with protocol addressing of the status/command area of the manager node, and an extended message body format, as specified below:

The protocol header specifies the word length of the message body.

The message body consists of one or more consecutive packets.

Each packet has the following format:

- Byte: Packet type
  - Byte: Word length of packet data (w)
  - Word: w words of data

Left byte	Right byte
Byte 0 Packet type	Byte 1 Word length
Data word 1	
Data word 2	
Data word 3	
Data word 4	
. . . . .	
. . . . .	
Data word w	

The format of the data words depends on the packet type. Three kinds are currently defined, where one or more kinds of data may be present for a given packet type:

Time: Consists of 6 or 8 bytes of BCD coded information according to DIN 19244:

- Byte: Second
  - Byte: Minute
  - Byte: Hour
  - Byte: Day
  - Byte: Month
  - Byte: Year
  - Byte: Status, bit(4) = 1: daylight-saving time, bit(0) = 1: keep time until next time packet
  - Byte: 10s or 100s of milliseconds

Second	Minute
Hour	Day
Month	Year
Status	msec

If the packet length is only 3 words then no status or millisecond bytes are expected. IGSS does not support daylight saving time and does not support the millisecond count. The time stamp remains in effect for the following packets in the message - and, if bit 0 of the status byte is set, also the packets in subsequent messages - until another time packet is encountered. If the time stamp packet is not present in a message, IGSS applies its own time to the received data.

Address: Used to relate the packet to a particular data area in the client (which may be the manager node itself):

- Byte: Node 0..255
- Byte: Data block 0..255
- Word: Word offset 0..4095

Node	Data Block
Word offset	

object values: The values that are to be assigned to the objects configured in IGSS. Seen from a communications point of view these data are transferred transparently. The interpretation of the data is specified in the "Packet Types" section.

The message body format is defined with the aim of providing an efficient means of communication for a large number of object values, while also keeping the format flexible enough for easy single object value transfer. The following modes are foreseen:

1) Communicating a single object value change:

- Time stamp packet (optional)
- Address and value packet

2) Communicating a number of changes from different sources:

- Time stamp packet (optional)
- Address and value packet (1st object)
- Address and value packet (2nd object)
- ...
- Address and value packet (nth object)

Optionally extra time packets can be interspersed if the times are different.

3) Communicating a large number of values from a single source:

- Time stamp packet (optional)

- Value packet: Address of data block area

A number of words defining many object values

The latter form will be particularly efficient in a start up or resynchronisation phase, where all object values must be sent to IGSS.

Chapter 4: External Data Types

4.1 External data types

Event-oriented communication supports external data types in the same manner as polled communication.

The external data types available for the individual driver are listed in the **System Configuration**program. For further details, click here.

These external data types can then be selected as part of the PLC address on the **Edit Mapping** tab of the object properties dialogue.

Chapter 5: Packet Types

5.1 Packet Types

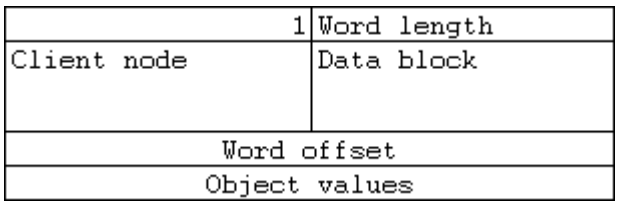
One or more packets are located in each message as described in "Message Protocols and Format". The packet types currently defined are listed below. The packet type is determined from the value of the first byte in the packet. Click on the packet type for further details.

1. Ordinary data packet
2. Message time packet
3. Request time synchronisation packet
4. Time synchronisation packet
5. System start packet
6. System reset packet
7. Keep alive packet
8. Alarm change packet
9. Communications error packet
10. Address List Request packet
11. Address List Response packet

5.2 Ordinary Data Packet

The data packet contains the start address as specified by node, DB, and word offset. All IGSS objects configured with addresses lying in the range from this start address to the end of the data area as defined by the packet length are defined to be part of the packet.

If IGSS receives the packet, then all these objects are evaluated and their new values are stored together with the
current time stamp.



If IGSS sends the packet, then it contains the current values of all objects defined in the area, except in the case where several digital commands are defined within one word. In this case, only the command that was actually issued is set - all other command bits in the word being binary zero.

5.3 Message Time Packet

The message time packet contains the time stamp defined as 6 or 8 bytes as specified in Message Protocols and Format. The current time for the message is set to this time. All values that are evaluated within this message until the next message time packet is encountered will be stamped with this time; however, if bit 0 of the status field is set, the time specified by the packet is used also for packets in following messages, until a new time packet is encountered.

Only bit 0 in the Status field is evaluated by IGSS; all other bits in this field are ignored.

If no message time stamp has been found in the message, IGSS will use its own system time to stamp the object values. If a message time stamp is invalid (eg seconds greater than 59) the following packets in the message - and, if bit 0 of the Status field is set, also those of following messages - are discarded by IGSS until a valid message time packet is encountered.

2	3, 4 or 5
Second	Minute
Hour	Day
Month	Year
Status	msec/10 (divided by 10)
msec	

Within IGSS each manager node can be configured to receive message time packets in all messages sent from IGSS. If so configured, each message from IGSS will contain exactly one message time packet as the first packet in each message.

5.4 Request Time Synchronisation Packet

3	0

This packet consists of the type and length byte only. The receiver should respond with a time synchronisation packet as soon as possible.

5.5 Time Synchronisation Packet

This packet contains the system time of the sender. This time is to be used to set the system time of the receiver. The synchronisation may be sent as a response to a request time synchronisation packet, or IGSS may send it at regular intervals to a manager node, if this node is so configured in IGSS.

5.6 System Start Packet

The system start packet contains the time when the system was last active.

IGSS will send this packet to all manager nodes that are so configured. These nodes are then expected to send all object changes that have occured since the specified time. Alternatively, they may send a copy of all object values, if this is more efficient.

5	3 or 4
Second	Minute
Hour	Day
Month	Year
Status	Msec

IGSS will ignore the reception of a system start packet.

5.7 System Reset Packet

The system reset packet contains a start address and a single data word.

When IGSS receives the reset packet, it will invalidate the values of a range of objects.

6	3
Client node	Data Block
Word offset	
Month	Year

Word length < 0:
The values of all objects configured in the specified Client node will be invalidated.

Word length = 0:
The values of all objects configured in the specified Client node and Data Block will be invalidated.

Word length > 0:
The values of all objects configured in the specified Client node and Data Block, from the specified Word Offset and Word length words forward, will be invalidated.

IGSS will not send any system reset packets.

5.8 Keep Alive Packet

The keep alive packet contains an address and a single data word. This packet is used to monitor whether nodes are running properly at the application level. Note that IGSS can only monitor directly connected nodes, and only at the communication level, unless keep alive packets are used.

7	3
Client node	Data Block
Word offset	
Heartbeat	

If an object is configured as a keep alive object for a given node, then IGSS will monitor this object for changes within the configured time interval. If no changes occur, then an alarm is generated for the object. The time interval is reset every time a keep alive packet having a value different from the current value is received.

If the keep alive object for a given node is also configured with an output component, then IGSS will send a keep alive packet at the configured frequency to the node. The heartbeat value transmitted will step through the configured range, being incremented by one unit each time.

5.9 Alarm Change Packet

An alarm priority level (pr_low) can be specified for each node, causing IGSS to send an alarm change packet to the node whenever the active and/or acknowledged alarm count changes for a priority at or above pr_low.

8	Word length
Client node	Data Block
Word offset	
#active hp	
#ack´ed hp	
#active hp-1	
#ack´ed hp-1	
.....	
.....	
#active pr_low	
#ack´ed pr_low	

An alarm is understood to be active when it is not ended and/or not acknowledged. The alarm change packet specifies, for each priority at or above pr_low, the number of active alarms, and how many of these alarms are acknowledged.

1 <= pr_low <= hp

hp: Highest alarm priority level

The low and high alarm priority levels are defined in the **System Configuration** program.

5.10 Communications Error Packet

This packet has two uses:

1) When a bad packet is received, a Communications Error packet returns the bad packet back to its originating application. In this case, a bad packet is in principle caused by the originating application and not by communication errors, as such errors should be detected and handled by the communications protocol.

In this case, the Error Group field must equal 45 (decimal).

A reason code specifies the reason for the packet being discarded, as follows:

1: Unknown type

2: Word length field is invalid, too long, or incompatible with the type field

3: The packet contains obviously nonsensical data (e.g. a message time packet specifying a month that is zero or greater than 12)

If the Word length field of the bad packet is in any way untrustworthy, the remainder of the message containing the bad packet is assumed to be part of the bad packet and is discarded.

9	Word length
Client node	Data Block
Word offset	
Error group	
Reason code	
Bad packet	

Error group = 45

2) The PLC application can generate a user-specified alarm in IGSS, with the field "Error group" specifying the alarm number.

9	4
Client node	Data Block
Word offset	
Error group	
Reason code	

30 <= Error group <= 39

5.11 Address List Request Packet

A node may request from IGSS a list of process I/O addresses used by IGSS. IGSS will respond with an Address List Response Packet.

IGSS will not send any Address List Request packets.

The List type field

The List type field specifies whether a list of nodes, data blocks or word offsets is requested:

List type = 0: List of nodes is requested.
The Node and Data Block fields are ignored.

List type = 1: List of Data Blocks is requested.
The Node field specifies the node for which the list of Data Blocks is requested.

List type = 2: List of Word Offsets is requested.
The Node and Data Block fields specify the node and Data Block for which the list of Word Offsets is requested.

The Access field

The Access field specifies whether the list is of input addresses, output addresses, or both:

Bit 0 set: Input addresses are requested.

Bit 1 set: Output addresses are requested.

Both bit 0 and bit 1 may be set; other bits are ignored. If neither bit 0 nor bit 1 is set, then the packet will be discarded and an Error Received packet will be returned.

10	2
Node	Data Block
List type	Access

5.12 Address List Response Packet

On receiving an Address List Request packet, IGSS will, as soon as possible, send an Address List Response packet. The Node, Data Block, List type and Access fields are identical to those in the corresponding received Address List Request packet. The Address List Response packet contains a bit string specifying which process I/O addresses are used by IGSS:

Examples

List type = 0 (list of nodes), bits 6, 41, and 206 in the bit stream are set: Nodes 6, 41, and 206 are used.

List type = 1 (list of DBs), Node = 6, bits 0, 100, 101 and 102 in the bit stream are set: DBs 0, 100, 101, and 102 in node 6 are used.

List type = 2 (list of word offsets), Node = 6, Data Block = 100, bits 0, 1, 10, 11, 12 and 13 in the bit stream are set: word offsets 0, 1, 10, 11, 12 and 13 in node 6 are used.

Note that the bit stream is in standard Intel format (in the blocks shown below, the least-significant byte is shown to the left).

Chapter 6: Reference and Lookup

6.1 Conventions in this Manual

The following typographical conventions are used:

Convention	Description	Example
User interface element	When referring to labels and names in the user interface.	The **Data Management** tab.
User input	When the user has to type specific data in IGSS.	Type the following description: Incoming flow in Tank 2
Module name	When referring to a module in IGSS	Open the **Definition** module.
Note	A note emphasizes or supplements important points of the main text. A note provides information that may apply only in special cases.	By default, the timestamp is in universal time format, UTC[1]. This can be changed in the Driver Log Filters dialog box.
Tip	A tip suggests alternative methods that may not be obvious in the user interface. A tip also helps the user in working more effectively with IGSS. A tip is not essential to the basic understanding of the text.	Alternative to this simple find function, you can also filter on text in the messages in Driver Log Filters dialog box.
Warning	A warning is an important note that is essential for the completion of a task. In some cases, disregarding a warning may result in undesirable functionality or loss of data.	If you disregard the System alarm, you may risk loss of data in the LOG and BCL files.

6.2 Getting Help in IGSS

IGSS comes with a comprehensive help system designed to help both system designers and operators to get started with IGSS as quickly as possible.

Documentation overview

[1]Universal Time Coordinated (formerly Greenwich Mean Time), used as the basis for calculating time in most parts of the world. IGSS uses this time format internally in the database. You can switch between UTC and local time by enabling or disabling the "UTC" field in various dialog boxes in the system.

The IGSS documentation includes the following items:

Documentation item	Description
Getting Started	An introduction to IGSS and its most fundamental terms and features. Getting Started is intended to get you up and running as fast as possible. The manual provides a system and architecture overview followed by a number of real-life use cases you can go through before building your first real IGSS project. The manual is available in Adobe Acrobat format (.pdf).
Module help	For each module there is a help file with the same name as the module itself, for example, Igss.chm for the Master module, Igss.exe. The help file is invoked by clicking the  in the upper right corner of the module. The Table of Contents will then allow you to browse through the topics.
Dialog box help☐	For each dialog box there is a help topic with the following standard information: • Overview • Preconditions • Where do I find it? • Field help Dialog box help is invoked by clicking the help button  in the upper right hand corner of the dialog box.
Thematic help	IGSS also provides thematic help. When there is a special theme that requires special attention from the user, a dedicated help file is provided. Examples include "Driver-Specific Help" and "Database Administration Help".

Where are the help files located?

The IGSS help files are located in the appropriate language folder under the [IGSS InstallPath]. The help files are available in English at release time.

The paths to the help files are:

Language	Path
English	[IGSS InstallPath]\ENG
Danish	[IGSS InstallPath]\DAN
German	[IGSS InstallPath]\DEU

Translated help files

Selected help files have been translated into Danish and German. If you require help files in your language, please contact 7T.

Help updates

The IGSS help files are continuously updated and improved. Check regularly with the **IGSS Update** module in the IGSS Start menu.

6.3 Version Information (IGSS Help System)

The IGSS help files are based on software build number 9212 (initial release)

English help files

To update the help files, you must activate the **IGSS Update** module in the IGSS Start menu. There must be a connection from the PC to the Internet. Every time **IGSS Update** is run, IGSS help files as well as IGSS system files will automatically be updated on the PC from the 7-Technologies web server.

You select the languages you want to update in the **Tools** menu of the **IGSS Update** module.

If you are not able to update the IGSS system directly via the Internet, the alternative is to download the updates from the 7-Technologies website as zip files. These can then be transferred onto a CD or USB memory stick, which is then the medium used to update on site.

> After running **IGSS Update**, the build numbers in various IGSS modules may change to a higher number. This signifies that the module in question has been updated with newer files. Build numbers consist of four digits, where the first digit represents the year and the last three represent the day number in the year in question. The build number can be seen in the **About** dialog box which can be activated from the **Help** menu.

An example:

Build number = 9212
9 = the year 2009
210 = The 210th day of the year

Chapter 7: Glossary

A

Application button

The Application button, nicknamed "doughnut", is located in the upper left corner of the module's window. Click the button to access the application menu. The menu contains items that were typically found in the File menu in previous versions of IGSS. In most modules, an "Options" item allows the user to define global module settings. The Application button is a new term introduced with the .NET platform and used in, for example, the Microsoft Office 2007 package.

atom

H

HDM

HDM is the abbreviation for Hour-Day-Month.

Q

Quick Access Bar

You can customize the Quick Access Bar to include the functions you use most frequently. Simply drag the relevant function from the ribbon to the Quick Access Bar.

R

Ribbon

The Ribbon is a new term/element in the Microsoft universe. The Ribbon replaces the well-known toolbars in applications. The Ribbon provides quick access to the most commonly used functions in the application. The Ribbon is divided into logical groups (the tabs) and each tab is divided into sections (the blocks in the tab). The Ribbon is context-sensitive which means that only relevant functions are accessible dependent on the current user action.

S

SCADA

Supervisory Control & Data Acquisition

U

UTC

Universal Time Coordinated (formerly Greenwich Mean Time), used as the basis for calculating time in most parts of the world. IGSS uses this time format internally in the database. You can switch between UTC and local time by enabling or disabling the "UTC" field in various dialog boxes in the system.